Automatic Hexahedral Mesh Generation on Many to Many Sweepable Volumes with Multiple Sweep Axes

Statement of Problem

Over the past two decades, Finite Element Analysis as emerged as in important tool for scientists and engineers. With this method answers to differential problems that would otherwise be unsolvable can be obtained. For three-dimensional problems there are two types of elements, tetrahedrons and hexahedrons. Although many automatic tetrahedral meshing algorithms that work on most geometry types have been developed, this is not the case for hexahedrons.

For solid modeling problems hexahedrons are the element of choice because they converge to the exact solution more quickly. Because there is no hexahedral meshing algorithm that works on all geometry types, the current approach to hexahedral meshing is to break the geometry is into primitives and use a different meshing scheme for each primitive. This approach is often referred to as a tool suite approach. Although this method has been successful in creating hexahedral meshes for numerous problems, it is very time-consuming, often taking weeks or even months for large problems. By automating this process for various collections of geometry primitives, the time for the meshing process can be greatly reduced.

Objective

The objective of this thesis is to create a new meshing algorithm for meshing two complex geometry types known as “multi-axis geometry” and “multiple-source/multiple-target geometry.” The meshing algorithm proposed by this research is a new approach to meshing these two geometry types and, with sufficient development, shows promise of being more reliable than any of the existing methods.

In order to create this new meshing algorithm, this research will borrow from two existing meshing algorithms known as “Sweeping” and “Grafting.” Because of the use of Sweeping and Grafting in this new algorithm, the new algorithm will be called the “GraftSweep Tool.”

Introduction to Sweeping

Sweeping is a hexahedral meshing algorithm that works on prismatic geometry types. The sweeping process is shown in Figure 1. In order to sweep a volume, all the surfaces of the volume must be classified as either a source surface, a linking surface, or a target surface. These Surfaces are shown in Figure 1a. Source and target surfaces are classified as those surfaces on the ends of the prism. If neither surface at the end of a prism is meshed it does not matter which is classified as the source or the target. However, if one of the surfaces is meshed then it must be classified as the source surface. Linking surfaces are those surfaces composing the sides of the prism. A requirement of all linking surfaces is that they must be able to be meshed in such a way that every interior node is connected to exactly four edges. This type of mesh is often referred to as a structured or mapped mesh and can be seen on the linking surfaces in Figure 1.

The next step in the sweeping process is to mesh the source and linking surfaces of the volume. Although the linking surfaces must be meshed with a structured mesh, this is not a requirement for the source surfaces. The structured mesh on the linking surfaces and the unstructured mesh on the source surface are shown in Figure 1b. Once these meshes have been created, a layer of hexahedrons can be produced by copying the mesh on the source surface to the layer directly above it and linking the layers together. This process is shown in figure 1c. Repeating this process for every layer of the volume produces the mesh shown in figure 1d.

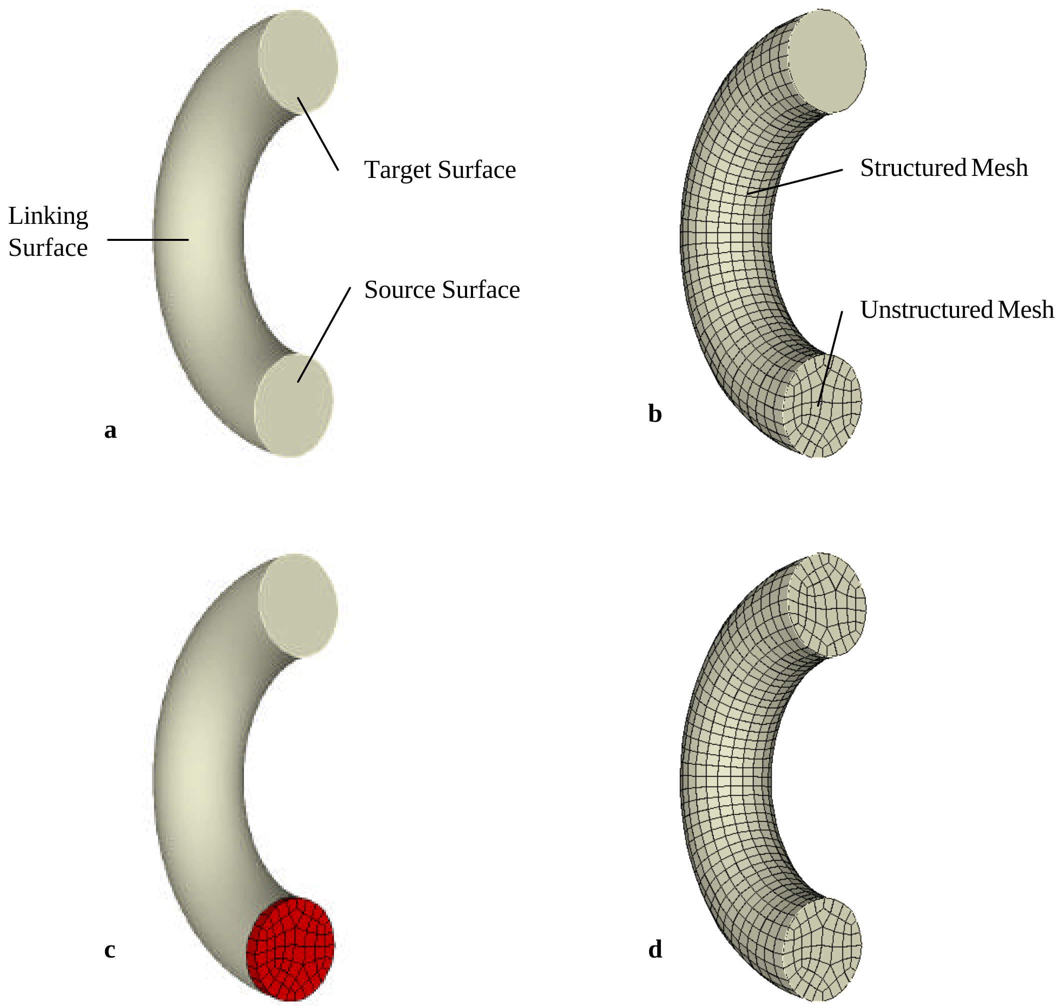


Figure 1. The sweeping process. **a**- The unmeshed volume showing source, linking, and target surfaces. **b**- The linking surfaces have been meshed with a structured mesh, and the source surface has been meshed with an unstructured mesh. **c**- The first layer of hexahedral elements has been formed, the linking surfaces are still meshed but the mesh is not shown. **d**- Meshing of the volume is complete.

Introduction to Grafting

Another important part of this thesis will be implementing the Grafting algorithm. The grafting algorithm was originally developed at BYU with the purpose of extending the capabilities of sweeping. It is a method of adjusting the mesh of one volume to conform to a second volume so that one continuous mesh can be produced throughout both volumes. This process is shown in Figure 2. In Figure 2a, two volumes, a cube and a cylinder, are shown. One is meshed; the other is not meshed. In Figure 2b, the surface mesh between the two volumes is shown. It is clear from this picture that the mesh on the cube does not conform to the boundary of the cylinder. To produce a continuous mesh between the two volumes this mesh needs be adjusted. In Figure 2c

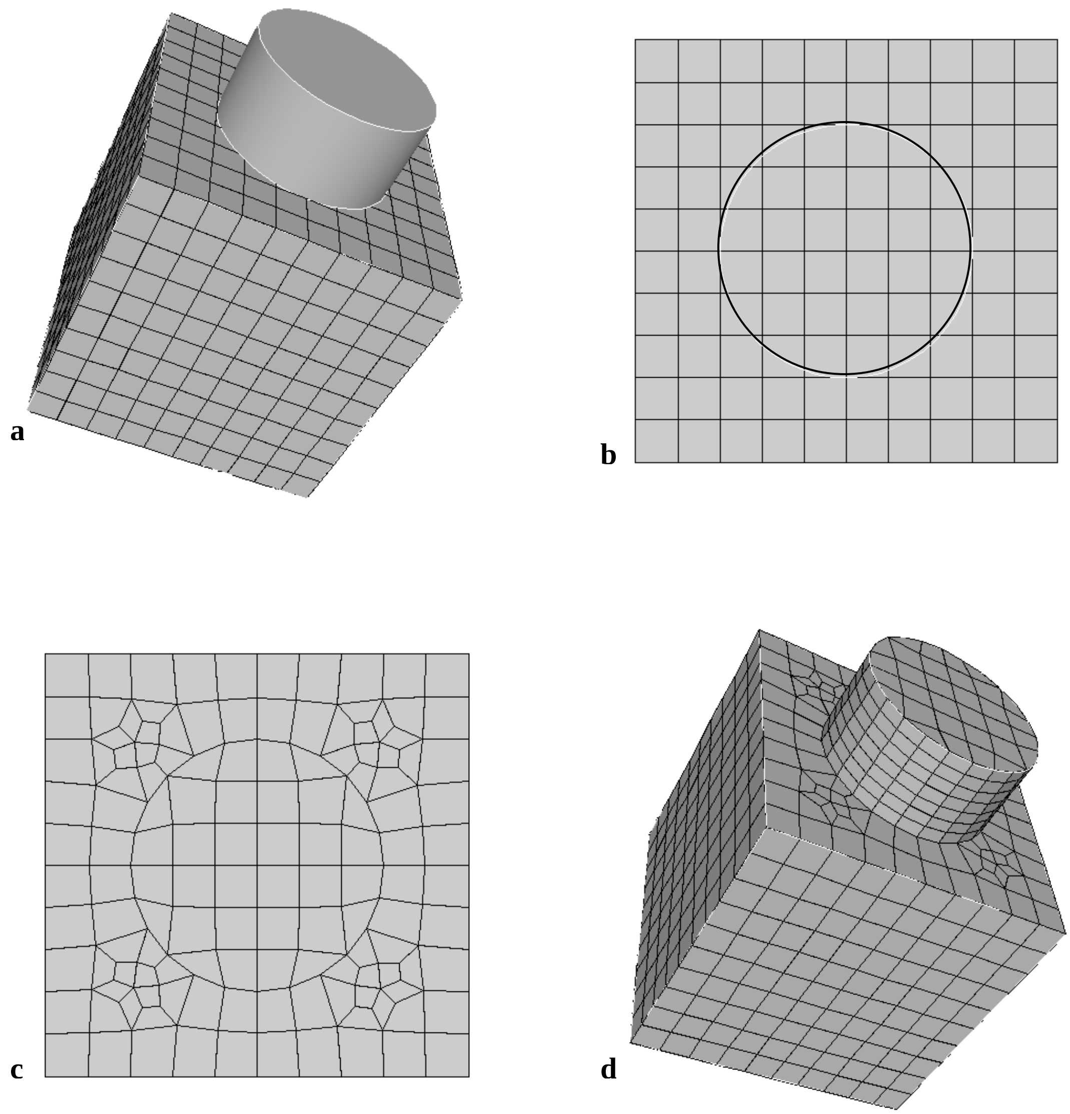


Figure 2. The grafting process. **a-**The meshed and unmeshed volumes are shown. **b**- The surface between the two volumes are shown. **c**- The mesh that has been adjusted through grafting is shown. **d**- The continuous mesh through both volumes is shown.

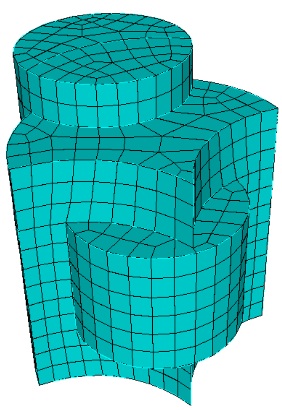
an adjusted mesh is shown. This mesh has been adjusted through grafting. The mesh under the unmeshed volume can now be swept through the volume to crate a continuous mesh. The final mesh is shown in 2d.

Analysis and Review of Current Work in the Field

The potential to use sweeping to mesh complex geometry types has been noticed by several others. Below is an explanation of four other attempts to use sweeping in a process similar to the one that I am proposing.

The Cooper Tool[1]

The Cooper Tool is a sweeping algorithm that has been designed to handle volumes with multiple source and multiple target surfaces. As shown above, the core sweeping algorithm can only handle one prismatic volume at a time. So, to sweep volumes with multiple source and target surfaces, the Cooper Tool breaks the source and target surfaces up into pieces that are geometrical matches to each other. Then, it creates internal links between matching source and target surfaces and uses these links in place of linking surfaces when creating the hexahedral mesh. This was one of the first tools that attempted automatic decomposition of a volume into prismatic primitives. The weakness of this tool however, is that it is not robust and often fails on the type of geometry for which it was designed. Part of this thesis will be an attempt to solve some of the robustness issues inherent in this algorithm. Geometry typical of what can be meshed with the Cooper Tool is shown in Figure 3.



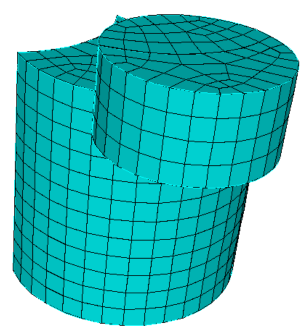


Figure 3. Example of geometry that is typical of what can by meshed with the Cooper Tool.

The MultiSweep Tool[2]

Another of the first attempts at mesh generation by breaking a volume into sweepable primitives is the MultiSweep Tool. This tool was originally developed at BYU. Its method of mesh generation is very similar to the Cooper Tool but there are two differences that are of note. Instead of decomposing both the source and target surfaces into matching pieces, the MultiSweep tool only decomposes the source surface. Because sweeping is a directional meshing scheme that starts from the source surfaces and ends at the target surfaces, this is not a problem. The second difference is that MultiSweep uses an algorithm to determine if any prismatic primitive is entirely interior to other prismatic primitives. In this manner the MultiSweep Tool can handle geometry with blind holes while the Cooper Tool cannot. A piece of geometry meshed with the MultiSweep Tool is shown in Figure 4.

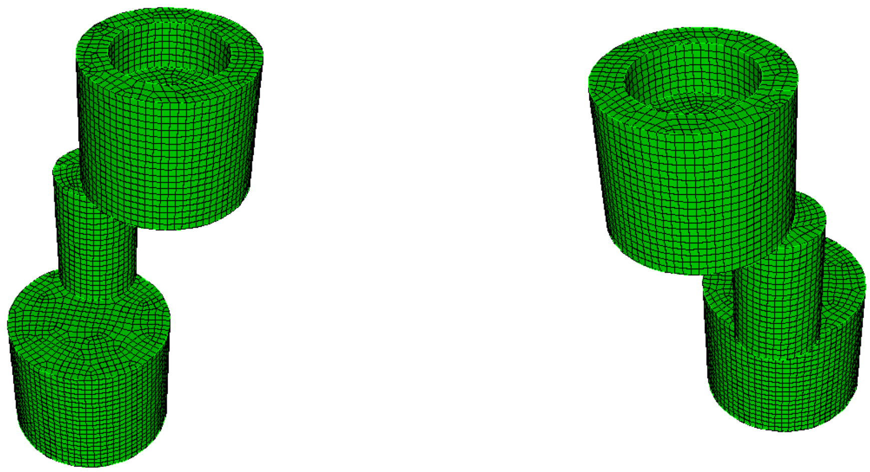


Figure 4. Example of geometry that has been meshed with the MultiSweep Tool.

The CCSweep Tool[3]

The CCSweep Tool is a new approach to handling geometry with multiple source and target surfaces. Although algorithms for meshing geometry with multiple source and target surfaces such as the Cooper and MultiSweep Tools have all had robustness problems, many robust algorithms for meshing volumes with multiple source surfaces and just one target surface have been developed. The strategy behind the CCSweep Tool is to take a CAD model and break it into prismatic pieces that can have multiple source surfaces but are limited to one target surface. Then any of the robust sweeping algorithms with the ability to handle multiple source surfaces can be used to mesh the geometry primitives. Geometry that has been decomposed using the CCSweep Tool is shown in Figure 5.

An important feature of the CCSweep Tool is that during the geometry decomposition process no actual changes are made to the CAD model. Instead the CCSweep Tool uses a type of virtual geometry used only by the meshing program. For the GraftSweep Tool, it is planned that some sections of the geometry models will have be separated in a manner similar to that of the CCSweep Tool. The GraftSweep Tool, in these instances, will use the same virtual geometry that is used by the CCSweep Tool.

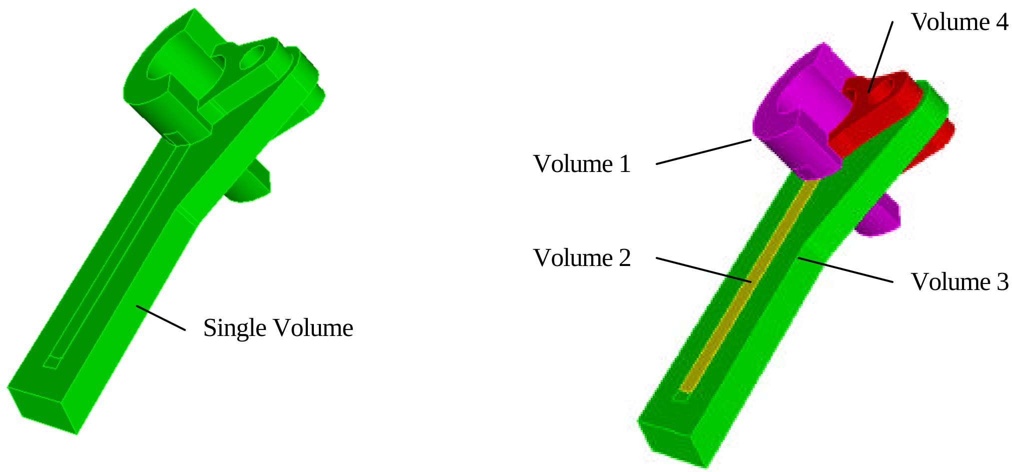


Figure 5. Geometry that has been decomposed by the CCSweep Tool.

The Multi-Axis Cooper Tool[4]

The Multi-Axis Cooper Tool is an extension of the Cooper Tool. Geometry that has been meshed with this scheme is shown in Figure 6. In this version of the Cooper Tool, geometry with multiple sweep axes is meshed by locating the protrusions on the linking surfaces and then removing the elements directly underneath the protrusion and remeshing this area to conform to the protrusion. This method works well on multiple axis geometry, but it requires that sections of the mesh on the source and target surfaces be meshed with a structured mesh. If a structured mesh cannot be created in these areas of the source and target surfaces, then multiple axis sweeping is not possible.

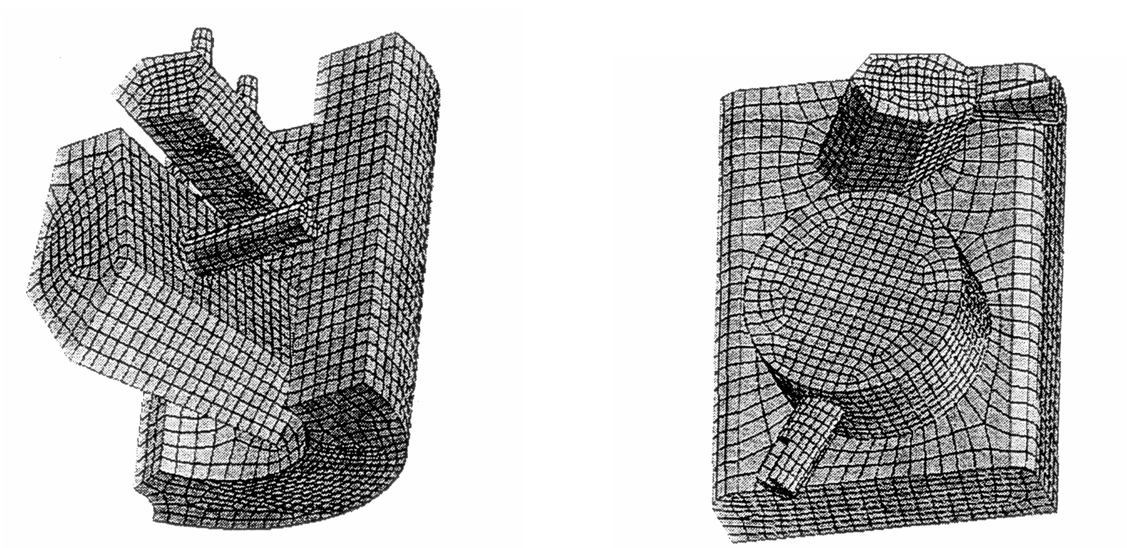


Figure 6[4]**.** Examples of geometry that have been meshed with the Multi-Axis Cooper Tool

Method to be Followed

The GraftSweep Tool combines grafting with ideas from the algorithms listed above. The method it will use for meshing is described in the following four steps.

Step 1 – Determine Sweep Axes

The surfaces of every object that is able to be meshed using the basic sweeping algorithm can be divided into sets of source, linking and target surfaces. The set of source and target surfaces give a direction for sweeping; this direction is known as the sweep axis. The surfaces and direction of the sweep axis for a cylindrical sweeping primitive are shown in Figure 7.

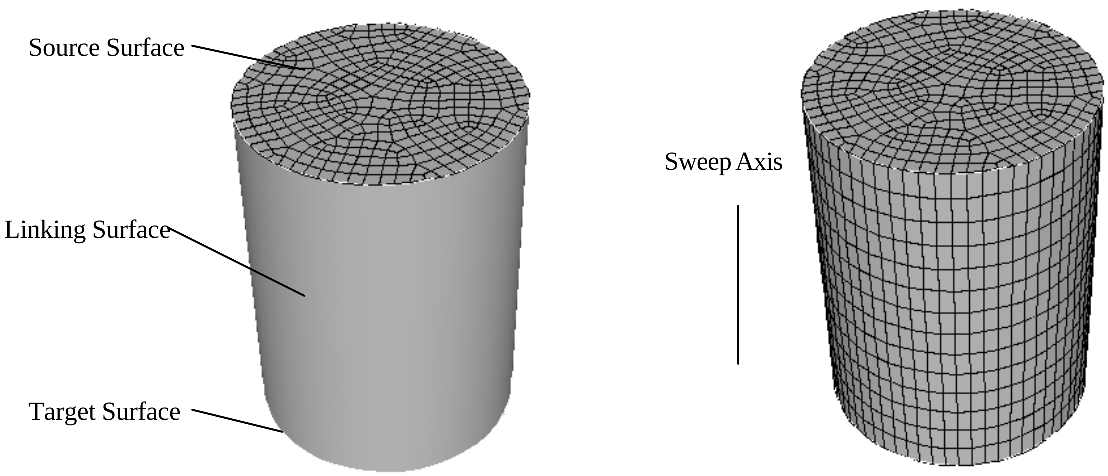


Figure 7. Cylindrical sweeping primitive. In the first figure, the source surface is meshed. In the second figure the volume has been meshed. The downward direction of the sweep axis also shown.

A key aspect of my research will be determining how to handle geometry that has multiple sweep axes. An example of such geometry is shown in Figure 8. In order to sweep this geometry, the section with the second sweep axis will need to be cut off. This will be done using the same type of virtual geometry as the CCSweep Tool. Once this has been done, the first cylinder can easily be meshed. Then the mesh from the first cylinder can be grafted onto the second cylinder and the second cylinder can be swept.

In order for this process to work, a method will have to be developed to determine which sections of the geometry need to be removed. The plan at this point is to have the user specify all the source and target surfaces of the volume. From this information the linking surfaces can easily be determined. Because every linking surface must be able to be meshed with a structured mesh, if any linking surface fails to mesh in this manner, it is an indication that there is a protrusion from that surface which needs to be removed. In this manner all the sweep axes of the entire volume can be located and meshed separately.

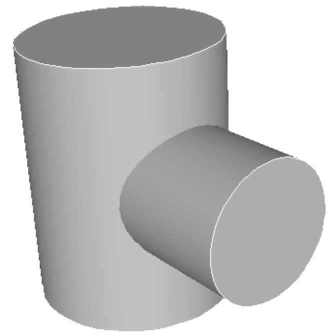


Figure 8. Example of Geometry with two sweep axes.

Step 2- Determine Layer Numbers

A common characteristic of all the geometry that the GraftSweep Tool is meant to handle is that it will have multiple source and target surfaces. In order to mesh this type of geometry, every source and target surface must be given a layer number corresponding to the sweep layer in which it resides. This is done by meshing all the linking surfaces and using that mesh to assign layer numbers to all the source and target surfaces. A recursive algorithm is used for the layer number assignment. This algorithm starts with a random source surface and assigns it a layer number of zero. Because the algorithm starts randomly, it is possible to have surfaces with negative layer numbers. For this reason, after all the surfaces have been layered, their layer numbers are normalized so that the lowest source surface has a layer number of zero. This layer information will then be used to determine the boundaries of each prismatic primitive. An example of geometry with multiple source and target surfaces is shown in Figure 9. This same geometry with meshed linking surfaces and layers assigned to the source and target surfaces is shown in Figure 10.

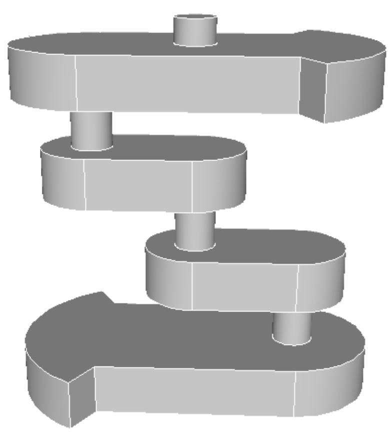


Figure 9. Geometry with multiple source and target surfaces

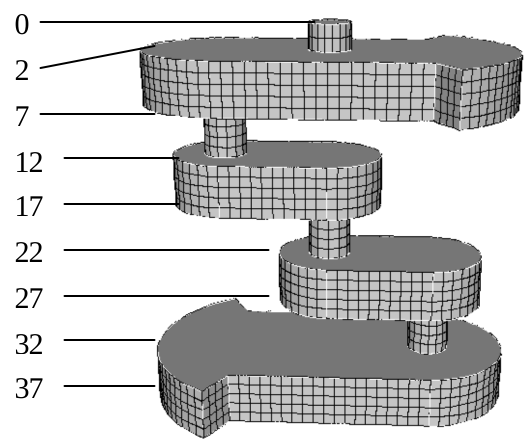


Figure 10. The same geometry with layers assigned to the source and target surfaces.

Step 3-Divide the Volume into Primitives and Mesh Them

Once layers have been assigned to each source and target surface the meshing process can begin. This is done by selecting the source surface with the lowest layer number that has not yet been meshed and sweeping until the next source or target surface is reached. If the next surface is a source surface, then the process starts over. If the next surface is a target surface, any curves that lay interior to the surface are grafted into the swept mesh and then the process starts over. When every prismatic primitive of the volume is meshed, the process is complete.

Step 4- Pillow if Necessary

One negative aspect of the Grafting algorithm is that it often produces poor quality elements. However, the quality of these elements can usually be improved by inserting a layer of new elements around them and then smoothing the mesh. This process is called pillowing. After all the hexahedral elements of the volume have been created, they will be checked for quality and pillowing will be performed if needed.

Delimitation of the problem

Automatic Surface Labeling

Currently the Sweep Tool, as implemented in Cubit, uses an algorithm for determining the source and target surfaces of a volume called the AutoSweepScheme. However, for multi axis problems the AutoSweepScheme will not be able to determine these surfaces. Development of a new AutoSweepScheme that will recognize these surfaces for multi axis geometries is an arduous task that will be left for future research.

Multi-Axis Sweeping

Producing a tool with the ability to mesh all geometry types with multiple sweep axes is not yet feasible. This is because of the complex nature of some of these types of geometry. For this reason this thesis is committed to handling geometry where each additional sweeping axis protrudes from a single linking surface. Additionally, when the protrusions exist on the boundary of the linking surface they are more difficult to detect. Methods for detecting such protrusions will be investigated, and attempts will be made to handle such types of geometry, however, this thesis will only be committed to the solution of problems where the protrusions are interior to the linking surfaces.

Problems with Pillowing

The most efficient way to pillow bad elements in the meshing process is to perform the pillowing operation before successive sweeping operations. However, due to quadrilateral elements that exist interior to the volume during the sweeping process, this is not possible. For this reason pillowing will be performed after the sweeping process for each volume.

Smoothing

Although pillowing improves the quality of the grafted mesh, poor quality elements can still result. Ideally these elements will be smoothed separate from the other elements in the volume. In this manner the smoothing process can be handled more quickly. However, it has not yet been discovered if smoothing in this manner is possible. If no simple method of smoothing in this manner is found, this thesis is not committed to developing such a smoothing scheme.

Contribution to be made by this thesis

This thesis will produce a hexahedral meshing algorithm that will work on geometry with multiple source and target surfaces and with multiple sweep axes. This algorithm will be more powerful than any previous algorithm that has used sweeping for mesh generation. Additionally, the quality issues that are inherent with grafting will be investigated. From this investigation it will be determined if the combination of grafting and sweeping can be used as a stand alone algorithm for meshing these geometry types, or if it would be better to combine these with some of existing geometry decomposition routines in future research.

References

1. Blacker, Ted, "The Cooper Tool", *5th International Meshing Roundtable*, Sandia National Laboratories, pp.217-228, October 1996.

2. Lai, Mingwu, Bezley, Steven and White, David, "Automated hexahedral mesh generation by generalized multiple source to multiple target sweeping", *International Journal for Numerical Methods in Engineering*, John Wiley, Vol 49, Num 1. pp.261-275.

3. DR. White, "Assesment, Metrics and Techniques For Hexahedral Finite Element Mesh Generation", Published Ph. D. Dissertation, Carnegie Mellon University, 2003.

4. Miyoshi, Katsuhiro, and Blacker, Ted, "Hexahedral Mesh Generation Using Multi-Axis Cooper Algorithm", *9th International Meshing Roundtable*, Sandia National Laboratories, pp.89-97, October 2000.